Mobile Learning Communities – Are We There Yet?

Professors: Klara Nahrstedt, Lawrence Angrave, Marco Caccamo, Roy Campbell, Brighten Godfrey, Indranil Gupta, Karrie Karahalios, Robin Kravets, Sam Kamin, Scott Poole, William Sanders

Research Assistants: Ying Huang, Zheng Zeng, Long Vu, Anjali Sridhar

Staff: Rick van Hook, Paula Welch

For Any Inquiries about the Report Contact: klara@illinois.edu

Technical Report, Information Trust Institute, University of Illinois at Urbana-Champaign, November 2010

Abstract

Advances in wireless technologies have allowed users to communicate ubiquitously anywhere and anytime and made it possible for users to access and exchange information through wireless hand-held devices such as cell-phones, tablets and laptops. Especially, if we consider our university campuses, 802.11 WiFi infrastructure is everywhere and most students use laptops and diversity of cell-phones for their personal and school work to connect to various resources via WiFi. In this report, we are exploring the premise of cell-phones as educational tools for undergraduate students in our educational institution, and creation of effective mobile learning communities through the usage of cell-phones. Through one year investigation, we wanted to answer three major questions: (1) how would a mobile phone be used as an educational tool in and out of classroom, (2) how would an educational unit deploy phones and educational services to all undergraduate students semester after semester, and (3) is the WiFi communication infrastructure at the educational institution ready for massive phone deployment in and out of classroom. Through combination of discussions with undergraduate students, implementation of educational projects by students and for students, deployment of phones in selected classes, and measurements of WiFi infrastructure performance during phone usage in classrooms, we present the complexity of the issues when answering the questions and the difficulties to consider when it comes to mobile learning communities on a university campus.

Keywords: mobile learning communities, educational applications on mobile phones, WiFi infrastructure for mobile applications, mobile phone user analysis, mobile learning on campuses

1. Introduction

We are seeing a ubiquitous deployment of mobile technologies such as cell-phones, tablets and laptops in our everyday lives. Especially, on our university campuses students use mobile technologies for their personal usage and access to various resources such as information about local events, restaurants, access to class websites, and sharing of local experiences with each other.

In this report we explore mobile learning community (MLC) at the University of Illinois at Urbana-Champaign (UIUC) campus, consisting of undergraduate students, WiFi-based infrastructure over which the students communicate, and our educational institution that provides education to students, (as shown in Figure 1). This mobile learning community (MLC) already uses WiFi 802.11 infrastructure everywhere on the UIUC campus to access educational servers and services via laptops and tablets. We want to explore the following question(s): **if we imagine that all undergraduate students of the MLC would get a mobile phone as an educational tool from the educational institution, would the university campus, its infrastructure, services, and people be ready for this step?**

Figure 1: *Mobile Learning Communities*

We have conducted one year study (August 2009 – August 2010) to find answer(s) to the above question(s). We have started to ask the questions within a smaller scale in our Department of Computer Science at the University of Illinois, Urbana-Champaign. Through the support of several industrial grants, and an infrastructure planning grant from National Science Foundation we explored to answer three major challenges embedded in the above question(s): (1) ***how would we use mobile phone*** as an educational tool in and out of classrooms, (2) ***how would we deploy the phones*** and educational services to all students, and (3) ***is our departmental infrastructure ready*** for large scale of phone deployments in and out of classroom?

Through multiple facets of investigation, including questionnaires to users, projects in selected classes, deployment of large scale phones in classes, and WiFi/software measurements and evaluation, we will describe the complexity of the answers to the above the challenges. We will show that there are still many problems that MLC groups still face. Furthermore, we will present lessons learned towards the next steps of effective mobile learning. In Section 2, we describe the methodology to explore answers to our three challenges. In Section 3, we elaborate on the 'potential usage' of mobile phones as educational tools in educational environment, coming from our students at UIUC and from students at Technical University (TU) Darmstadt. In Section 4, we present educational projects that students from the distributed systems class, cs425, designed and developed during Fall 2009, as an effort to build educational services for students by students. In Section 5, we present the development of Slice software on the Android phone that provides pen-writing interface and capabilities. In Section 6, we describe the educational toolkit software, called mi-clicker, that has been developed in cs425 class and further hardened by a group of undergraduate students in Spring 2010. The goal was a major deployment of phones and mi-clicker application in another undergraduate class, cs241. Section 7 presents the deployment efforts in cs241 class. Section 8 and 9 provide student evaluation of the deployment efforts in cs241 as well as WiFi infrastructure measurements during the deployment efforts in cs241 class. Section 10 discusses related work in this area. We conclude our answers to the three challenges in Section 11 with further lessons learned and next steps.

2. Methodology of Investigation

To understand mobile learning communities and finding out how cell-phones hold as educational tools in our undergraduate classrooms, we have considered a mixture of approaches to find answers to the three challenges/questions: (Question 1) what should be the educational tools on our phones, (Question 2) how would we deploy phones and services to all undergraduate students, and (Question 3) is our departmental infrastructure ready for MLC.

Methodology to answer Question 1: To explore the answer to the first question, we have gone to the *potential users*, our students and asked them *via questionnaires* what educational tools they would like to have in and out of classroom if each of them would have a phone and used it, e.g., in classroom.

The second approach was to fold the answer into *projects* within the *distributed systems class, cs425, in Fall 2009.* The students in this class received each an Android Google phone and were asked to build in groups an educational tool on the phones that they wished would have as undergraduate students. So the challenge was to build educational services for the students by the students.

The third approach was to actually *deploy phones on a larger scale in an undergraduate class*, the cs241 in Spring 2010, install some of the educational services from the class cs425 on the phones and have students use the phones and educational tools through part of the Spring 2010 semester.

The forth approach was to get 'exit' feedback from students who used the phones and educational services in the undergraduate class.

Methodology to answer Question 2: To understand the process of deployment of phones as educational tools, and educational services, we have experimented with two major classes, the cs 425 class in Fall 2009 where we have deployed phones as development tools to each student in the class, and the cs241 in Spring 2010 where we have deployed phones on a voluntary basis as educational usage tools to as many students as wanted to participate.

Methodology to answer Question 3: One of the most important issues was to investigate performance of phones as educational tools when clustered in a large classroom, such as 1404 Siebel center, which seats up to 200 students. We have conducted two types of measurements: (a) large scale in the classroom during the Spring Break, and (b) small scale in test labs throughout the semesters.

3. Potential Users – 'Entry' Questionnaire

Mobile phones have been used for personal usage and are ubiquitous in their usage anywhere and anytime. However, as students enter the age of mobile learning communities, we wanted to ask students how they would envision using phones as educational tools inside and outside of classroom. We have prepared an anonymous questionnaire and distributed the questionnaires within two undergraduate classes: (1) the "Systems Programming Class", cs241 class in Fall 2009 (taught by Prof. Marco Caccamo), and (2) the "Introduction to Communication Networks", Comm I class in Summer 2010 (taught by Prof. Ralf Steinmetz). We have received 40 responses from the cs241 class and 29 responses from the Comm I class.

The questions have been grouped in three categories: (a) what kind of potential educational usage of phones do the students see **inside and outside of classrooms**, (b) what kind of potential usage of phones do the students see with respect to **better communication with instructor and teaching assistant**s, and (c) what kind of potential usage of phones do the students see in **group projects and communication with classmates in general**. The summary of most frequent responses is in Table 1. The detailed questionnaire is attached in Appendix 1.

3.1. Responses from Illinois cs241 Class

The cs241 class in Fall 2009 had undergraduate sophomore and junior students who had some exposure to wireless technology in classrooms prior to the cs241 class, such as usage of i-clicker in the Physics classes, and online access to web-based material, quizzes through the web, and usage of web-technology by the instructors. Hence, the students had a very rich set of responses and it was very obvious that they thought carefully about extensions of some of the online educational tools that they have seen before.

The most frequent answers to the first category of questions (educational tools for in/out of classroom) were: (1) ***advanced quiz application*** (similar but better i-clicker application) from the phone, (2) ***notifications about due dates, exams and tests***, (3) ***access to class material*** on the phone such as access to slides, home page of a class, and (4) ***access to Wikipedia and dictionary of basic concepts*** during lectures.

The most frequent answers to the second category of questions (communication tools with instructors and teaching assistants) were: (1) ***chat application*** with teaching assistant and instructor during and after lecture. Especially, students saw needs to ask teaching assistants questions during lectures and ***get quick clarification***. The students also would like ***to give feedback to instructors*** and let them know if the lecture is going too fast or too slow. (2) ***phone client software for newsgroups, email, instant messaging, ssh, text editor, gcc*** to communicate with instructors, teaching assistant and others more efficiently.

The most frequent answers to the third category of questions (group communication tools) were: (1) ***group notification capabilities*** about joint meetings, (2) ***group chat***, (3) ***group access*** to SVN tools and ***notification of updates,*** and (4) ***group meeting schedules.***

3.2. Responses from TU Darmstadt Comm I Class

The students in Comm I class consisted mostly of sophomore and junior students and they have been exposed to online educational technologies such as wiki, video recordings of their lectures, homework submission system and others. Hence, we have again seen impact of their prior knowledge and suggestions of existing improvements and new features when using phones in educational process.

However, as it will be clear from the answers, since these students have not been exposed to i-clicker technology in their prior classes, this application does not appear among the answers.

The most frequent answers to the first category of questions (educational tools for in/out of classroom) were: (1) ***access to class material*** such as slides, home pages, and video recordings, (2) ***access to Wikipedia***, (3) ***live streaming of lectures and podcasting***.

The most frequent answers to the second category of questions (communication with instructors and teaching assistants) were: (1**) *chat with teaching assistant and instructor*** in order to ask questions for quick feedback and vote to give feedback to instructor about quality and speed of the lecture, (2) ***applets and animations clarifying lecture material***, (3***) translation services*** among different languages, (4) ***update-feeds*** that announce new uploads of exercises and lectures.

The most frequent answers to the third category of questions (group communication) were: (1) ***group access to SVN*** (version control of group project software), (2) ***group scheduling service*** like doodle.com, and (3) ***group archiving capabilities***.

As we can see, students from both classes have many overlapping application ideas, such as having tools to access class material from phones, have chat capabilities with teaching assistants and instructors, especially during lectures to ask quick questions, but there are also differences based on the prior experience (e.g., the knowledge about i-clicker) and diverse knowledge of English by German and other foreign students (the Comm I class is being taught in English) as the suggested translating service indicates.

Summary of responses from Illinois	**Summary of responses from TU Darmstadt**
Advanced quiz application (advanced i-clicker)	Access to class material (slides, home pages, video recordings)
Notification about due dates, exams, tests	Access to Wikipedia during lectures
Access to class material (slides, home page)	Live streaming of lectures and podcasting on phones
Access to Wikipedia and dictionary during lectures	Chat with teaching assistant and instructor (vote to give feedback about quality/speed of lecture, ask questions with quick feedback)
Chat with teaching assistant and instructor to ask questions, get quick clarification, give feedback to instructor about going too fast/slow	Provide applets and animations clarifying lecture material
Need phone clients for newsgroup, email, instant messaging, ssh, text editor, gcc	Translating services among different languages
Group notification	Update-feeds that announce new uploads of exercises and lectures
Group chat	Group SVN access
Group SVN access and notification of software updates	Group scheduling service
Group meeting schedule	Group archiving capabilities

Table 1: *Summary of Responses from "Potential Users" of Phones as Educational Tools at Illinois and TU Darmstadt*

4. CS 425 Projects

In Fall 2009, we have purchased 40+ G1 Android phones through the Vodafone equipment grant that was available for educational purposes. The goal was to deploy the phones in a class which would develop educational tools for undergraduate students.

The reason we have decided on the purchase of Google phones was three-folded:

(1) the phones are WiFi-phones and do not need any data or voice services from any telecommunication service provider to start working, i.e., the students can use the phones and communicate with each other and with educational services of the department via the 802.11 wireless infrastructure; (Note: this requirement was very important to us since we are not allowed to pay for any student phone-data/voice services if we give students phones for educational purposes. This means that we needed phones that can be started/bootstrapped directly via WiFi – infrastructure provided by the educational institution.)

(2) Android has a simple and powerful SDK with detailed online documentation and tutorials. Due to open-source origin, there is a thriving developer community with many available third-party libraries. In addition, the development environment for Google phones is the generally well-known Eclipse tool which runs on any Linux, Windows, or MAC OS computing platforms

(3) importantly, there was no licensing, distribution or development fee. There was no charge to download and use programming tools, which becomes problematic in a large classroom for some commercial products. There was no charge to publish educational applications in Google Market. This fact eased the application installation and increased our project visibility.

(4) the programming language of Google phone application is a version of Java, hence the startup time of developing applications on phones was very fast since our introductory courses teach Java and students are familiar with the syntax and development environment.

We have selected the Distributed Systems class, cs425, taught by Prof. Klara Nahrstedt, and teaching assistant Ying Huang. The student population of cs425 class consisted of senior undergraduate students and first year graduate students. The class content was tested on a set of machine problems that exposed students to distributed system concepts. In Fall 2009, the students in groups of 2-3 worked on three machine problems that led to building educational tools for mobile phones such as G1 Android phones. The machine problems have been developed in the student programming laboratory, residing in the basement of the Siebel Center. The first two machine problems have been developed using the general development tool, called Eclipse. The last machine problem has been using the G1 Android phones. The machine problems entailed the following parts:

- In the first machine problem (MP1), the students created a simple chat service where two phones would exchange simple 140 bytes messages and allow a user to open one-to-one chat sessions with multiple users using the 802.11 wireless departmental infrastructure. To achieve this functionality, the students learned about distributed algorithms, services and protocols to achieve peer-to-peer membership, registr**at**ion and discovery protocols, monitoring algorithms of consistent states, handling of peer failures and search capabilities for peers.

- In the second machine problem (MP2), the students created a distributed file service where phones/users were able to locate files from each other and download/upload files among each other in a peer-to-peer manner. The students learned further distributed concepts of consistency management among distributed files and failure handling not only due to file transmission failures but also due to mobility of peers.
- The third machine problem (MP3) was open. The main goal was to use any of the distributed services and protocols from MP1 and MP2 and develop an educational tool that younger undergraduate students would like to use. We asked the cs425 student to think of educational services ***they would have liked to have when they were freshmen, sophomore or junior students***. This assignment was to develop educational applications by the students and for the student. Also, we encourage students to develop features utilizing uniqueness of phones, such as GPS, WiFi signals, accelerometers. This encouragement diverted the students far away from traditional machine problems design based on desktops and laptops. Students showed great passion on designing novel distributed applications. The last MP3 was also announced as an open project that would be demonstrated on the phones and judged by a group of faculty and researcher from ***Qualcomm, Dave Craig*** (Alumnus of ECE UIUC). Qualcomm was very interested in the outcome of the educational tools and offered Kindle devices to the members of the winning team (team with the best educational application on G1 phones).

It is important to stress that most of the final cs425 MP3 educational applications have considered the phones as clients and a central web server assisted in maintaining peer membership list, database of group members, quiz questions, notifications and other state information. Non-state data and messages are exchanged among phones in peer-to-peer fashion. All networking between phone clients and client-server has been conducted via WiFi 802.11 networking as shown in Figure 2. In the next subsections, we will describe three of the many application services that the cs425 class built in the MP3 assignment. It is also important to mention that the staff of cs425 class provided feedback to cs425 students from the cs241 Fall 2009 questionnaire, i.e., the answers that cs241 students indicated and are shown in Table 1, but the cs425 students were also encouraged to think about new applications.

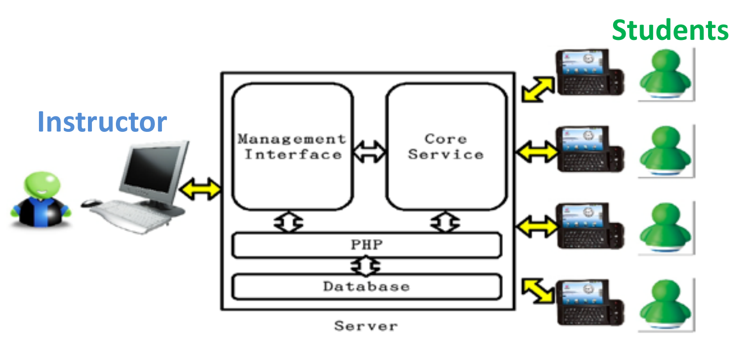

Figure 2: *Main Architectural Design of Many of the Educational Services in CS 425*.

4.1. G-Clicker Educational Application

The winning educational application was the ***G-Clicker*** educational application, a major extension to the i-clicker software for phones. The G-clicker tool has several services bundled together as follows: (1) announcement capability to allow instructor to announce dates for machine problems, exams and any other updates, (2) exchange of course material, especially the ppt slides and pdf files, (3) quiz taking within the classroom, (4) getting attention in a big class via phone gesture recognition, and (5) voice recording to leave voice message for a teaching assistant. Figure 3 shows the user interface on the phones for G-clicker. The phone gesture recognition is especially an interesting feature, where using phone accelerometer, student shakes the phone during the lecture (if he/she has a question), which causes creation of a message sent to the instructor's computer, and the instructor sees that somebody has a question. The instructor can pause and ask if there are any questions. This might be especially useful in large classrooms where the instructor might not see in the back of the classroom if anybody raised a hand and has a question.

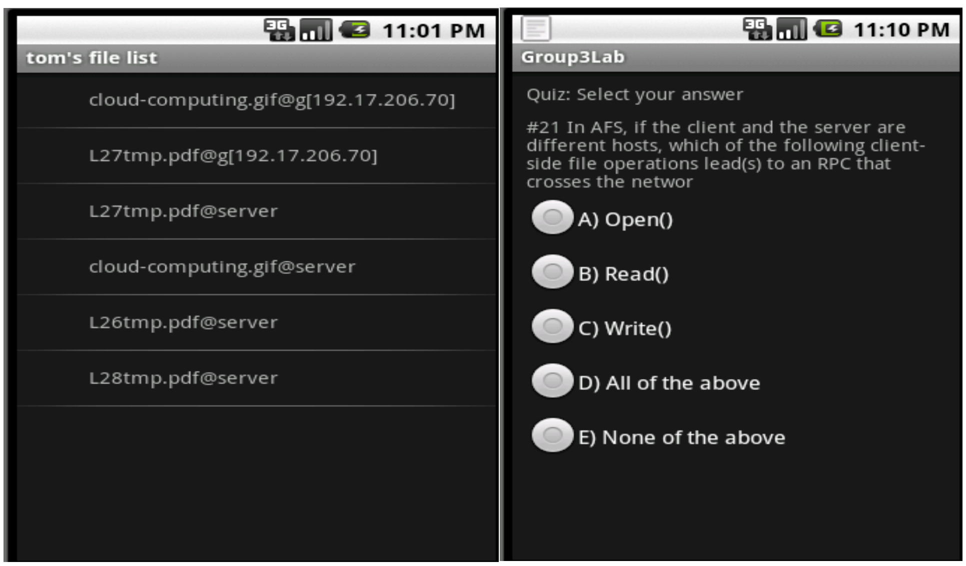

Figure 3: ***G-Clicker*** *User Interfaces for (left) file exchange, and (right) quiz services.*

4.2. WhereAmI Educational Application

Second educational service in the cs425 class competition was the '***WhereAmI***' service to help the freshmen students. The goal was to give freshmen students a mobile phone application that allows them to orient themselves in the Siebel Center. The application represents an augmented reality where video from the phone is overlayed with tags indicating if students are in the launch room, in the student lab, or in what classroom, and what direction other rooms of interest are. This application provided out-of-classroom virtual tour guide, based on existing indoor navigation algorithms (the students stored the RSSI map of the Siebel Center and used it) to locate rooms and people such as teaching assistants (and their offices). They also stored schematic graphical maps of individual floors to assist in student navigation. Figure 4 shows the user interfaces of this educational service.

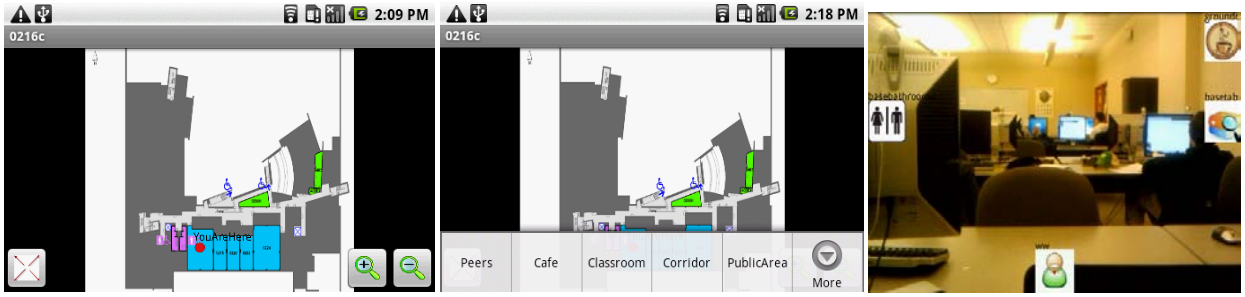

Figure 4: ‘***WhereAmI***’ *User Interfaces showing (left) the basement floor with location of the virtual tour guide holder (red dot) and the rooms floor plan, (center) the menu capability to highlight, for example, locations of labs only, or public areas only, and (right) video of the student lab tagged with directions to other rooms (e.g., restroom, coffee-shop).*

4.3. CalTalk Educational Application

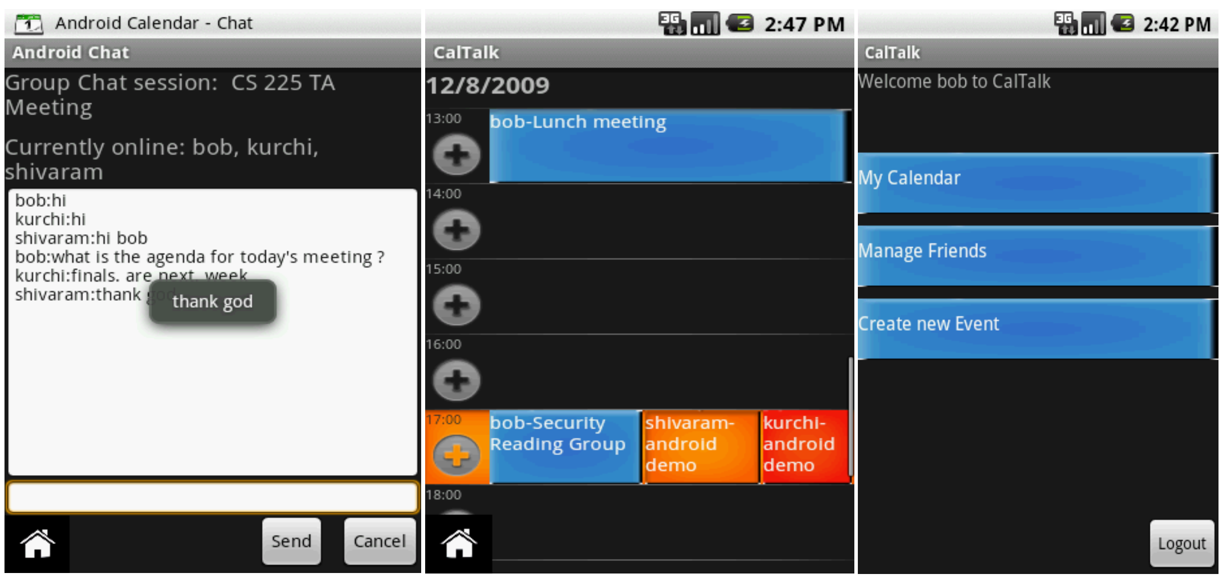

Figure 5: ‘***CalTalk’*** *Group Management System with User Interfaces showing (left) chat capabilities, (center) calendar entries, and (right) personal management of calendar, friends and events*.

The ‘***CalTalk***’ educational application was ranked third in the cs425 class competition and the goal of this application service was to help students with group projects and group management. The CalTalk included the following services: (a) multi-phone calendar collaboration, (b) a personal calendar view, (c) provision to add and delete events that impacted group members, (d) management system based on the ‘friend’ concept, and (e) group chat system. Figure 5 illustrates the phone user interfaces to access the group management system. This application provided extensive services for out-of-classroom cooperation among students. Especially, the introduction of the concept ‘friend’ from the other social networks tools has been welcome since it was acknowledged that even within group of students working on a joined MP, subgroup of friends might exist and talk about other issues.

4.4. Lessons Learned and Recommendations from cs425 MPs

Throughout the designing and executing process of three MPs, we made the following observations, regarding NAT, battery, debugging and pressure on wireless infrastructure.

First, unlike lab machines, phones communicate by internal IP address, which restricts the scope of peer-to-peer communication within a subnet. Coupled with firewall configuration in different APs on campus environment, the experiments could be only carried out when all the phones were associated with the same AP. Also the server should either own a public IP or be associated with the same AP as phones.

Second, battery was a big concern. During demo, phones died quickly and needed to be continuously charged. Students focused more on functionality design and ignored other resource issues, energy consumption in particular. We did not expect them to address those issues in the distributed system class.

Third, peer-to-peer debugging was troublesome in Android, if peers listened to incoming traffic at a random port. This was due to the tedious port forwarding setup step, at the time when MPs were designed. It is recommended that students test networking and computation components separately before proceeding to the system-wide testing.

Finally, during demo, we needed to limit the WiFi activities from non-demo devices. With wireless infrastructure optimized for laptop access, phone encountered numerous WiFi association and communication problems, when the whole class was coding and testing in the same lab, waiting for their turn to demo. This situation gained us knowledge about whether students deal with all kinds of network exceptions nicely. However, sometimes it was difficult for us to tell whether it was network problem or implementation problem. It even slowed down our demo schedule, since we needed to wait for phones to get associated and live again. In the future, instructors and TA who will use Android phones in classes, they should control the number of extra students in the lab, besides the students who show the demo.

5. Android Phone Efforts for Writing Capabilities

One affordance of the Android phones which we want to explore further is **writing**. The writing capabilities of these phones are crude, but still potentially useful. Writing is a much richer communication modality than typing or button-clicking; this is particularly true when it comes to technical material, which often involves mathematical formulas that are not easily typed, or drawings that cannot be typed at all.

To explore the potential of writing on the phones, Prof. Sam Kamin (kamin@illinois.edu) and his research assistant Wade Fagen have ported the Slice system to Android. This system (slice.cs.illinois.edu) has been in development for use on Tablet PCs for several years, and has been used in a variety of experiments as shown in Figure 6. Slice is a system which combines an underlying core, originally written in C#, with a powerful extension layer, originally using Python. Entire applications can be written at the scripting level. One application in current use is a lecture application, which allows the instructor to “untether” her tablet from the projection system; an enhancement we are experimenting with gives students tablets and allows the instructor to see what they are writing. Another, entirely different application – but still implemented solely in the script layer – is a code review application, in which students can discuss code by each writing on a Tablet PC as the code author explains what they are reading; this has been successfully tested in our Programming Studio course and we plan to extend it to all sections of that course.

The Android phones had neither a satisfactory C# nor Python implementation, so the porting effort was quite significant. We have rewritten the core of Slice in Java, and have implemented a new scripting language from scratch, so that we would have perfect compatibility between the Android and Tablet PC platforms.

With this port, we are able to script applications that will run across Tablet PCs, Windows-based workstations, and Android phones, the only differences being those required by the differing form factors. We believe this will greatly simplify the process of developing applications for classrooms based on a variety of devices. In addition to the Android phones, we hope to obtain Android-based tablets for our experiments.

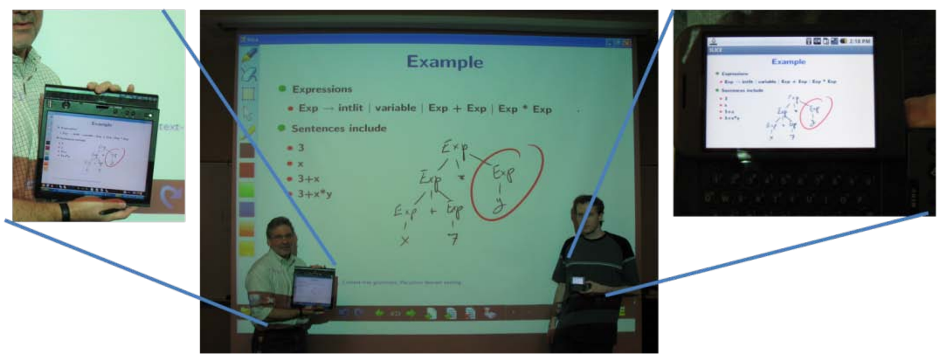

Figure 6: *Slice Implementation on Tablet PC*

Our first experiments, planned for the Spring 2011 semester, will use the phones as advanced “clicker” devices – like mi-clicker, but with the inclusion of writing. The students will be able to respond to short answer, rather than just multiple-choice, questions. Rather than viewing a histogram of the results, the teacher will be able to review all responses on a large display (the “dashboard”). (We note that short answer questions are potentially far more valuable to a teacher than multiple-choice; that latter can only confirm what a teacher suspects, while the former can actually show the teacher something unexpected.)

6. mi-clicker – Students for Students Efforts

One major result of the cs425 class was that several undergraduate students wanted to continue with refining some of the educational tools and wanted to see the actual deployment of the educational tools. We all understand that usually class projects work during the actual demonstration but do not scale to an actual deployment in a real class. Hence, with the group of enthusiastic CS and ECE graduate and undergraduate students, and under guidance of Professors Nahrstedt and Angrave, we have picked few services from the cs425 class projects and started to 'harden' them towards a real deployment experiments in Spring 2010. Out of this effort, the software '***mi-clicker'*** (***Mobile Illinois Clicker)***, was created.

6.1. mi-clicker Functions and Architecture

Based on the popularity of the enhanced i-clicker in the cs241 survey and also due to the winning G-Clicker educational application in cs425, we have decided to continue the work on 'clicker-like' set of educational tools (tool-kit for inside of the classroom) and deploy/test them in a classroom. mi-clicker includes the following functions: (a) 'take quiz' function, (b) 'get course announcements' (MPs, homework, exams) function, (c) 'text message with teaching assistant and/or instructor', and (d) 'ask question via audio' to teaching assistant and/or instructor.

The mi-clicker basic service architecture is web-based and client-server based, where the clients are phones and laptops. The clients on the phones will have interfaces used by students, the client on the laptop will have interface for the teaching assistant/instructor as shown in Figure 2.

The server architecture is a web server with (1) extensive database (mySQL) keeping information about the class students that are registered in the particular class, quiz questions, quiz results, files and any other information related to mi-clicker content, (2) PHP interface between the secure database and the core services and management interface, (3) management interface that serves the instructor/teaching assistant to enter quiz questions, announcements, see students answers, get text messages, questions, and provide text for answers, and (4) core services that are responsible for distribution of quizzes, announcements to students, receive answers, receive text/audio questions and store them safely in the database via PHP interface. It is important to stress that the functional part storing student database has been stored on the secure departmental server during the experiment, described in Section 7 and 8. The students and instructor had to authenticate to get access to the server and receive announcements, conduct quizzes and leave text/audio messages for instructors/teaching assistants.

6.2. mi-clicker User Interfaces

The educational tool mi-clicker has two major interfaces: (a) the instructor side and (b) the student side.

Instructor side course website includes capabilities to create announcements, import class enrollments, submit quizzes, check for answers, and check for voice mails and text messages. With respect to quizzes, the anticipation is that the instructor gives impromptu quizzes in the class to see if the class is following the material. Hence, the capability is built in to create a summary of answers for the instructor to see what portion of class understands the particular material. Figure 7 shows an example of the instructor course website.

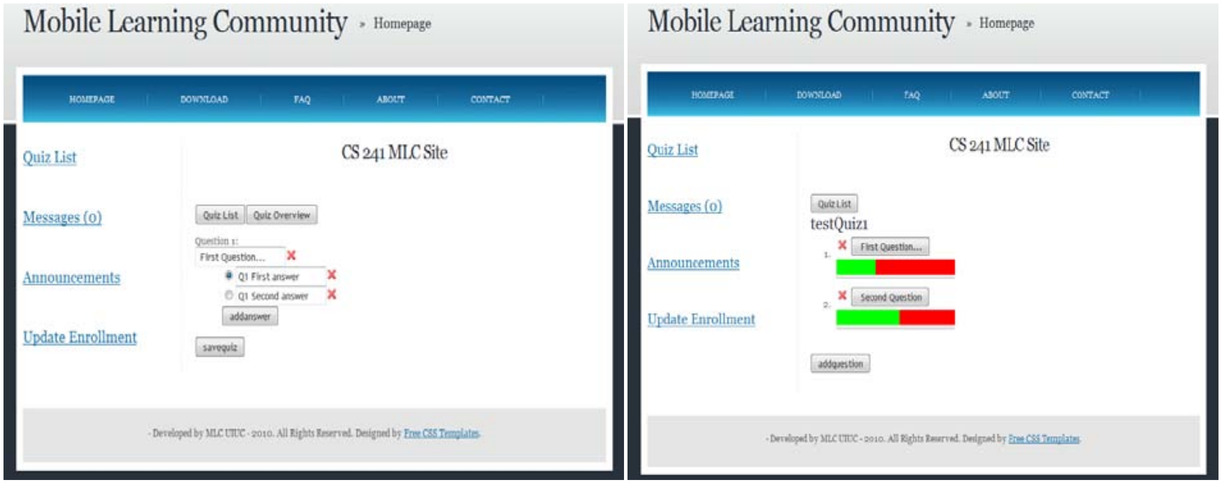

Figure 7: *mi-clicker Instructor Side Interface showing (left) Quiz Management, and (right) Quiz Summary*.

At the **student side**, the student interface allows the student to authenticate and then access four different services, the quiz service, the announcement service, the text messaging service and the voice service as Figure 8 shows.

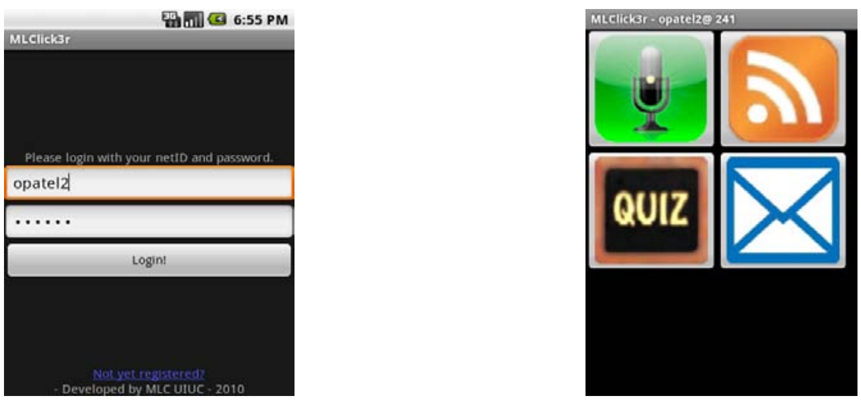

Figure 8: mi-*clicker Student Side Interfaces with (left) Authentication, and (right) Four Services*

The quiz service presents the currently running quiz with questions and options, as well as information if the quiz was submitted (see Figure 9-left). The announcement service shows short announcements as well as information if announcements are new, read, and at what time they appeared (see Figure 9-center). The text messaging service allows the students to send short questions to the teaching assistant during the lecture (see Figure 9-right1). The voice recorder allows recording of voice question, replay, and sending it to instructor or deleting the recording (see Figure 9-right2).

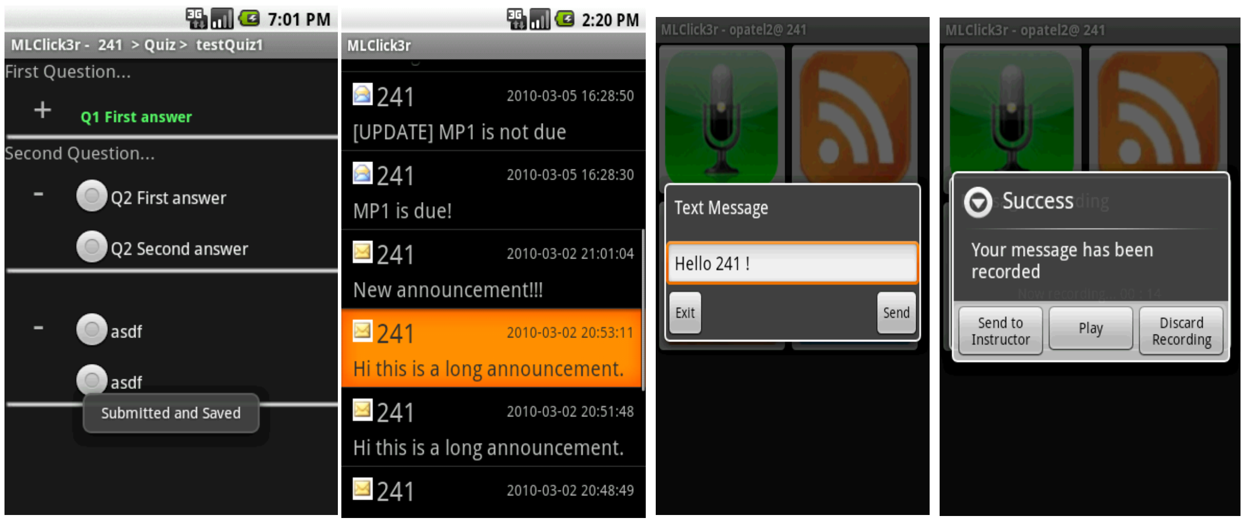

Figure 9: *mi-clicker Student Side Interfaces of (left) Quiz, (center) Announcements, (right1) Text Messaging and (right2) Voice Recording*.

6.3. mi-clicker Technical Details

The mi-clicker services have been implemented using Java on Android G2 phones which we have purchased via the NSF CNS IT grant and tested out of the classroom deployment (prior to actual class deployment) as described in Section 9 as well as in classroom deployment as described in Section 7 and 8. The 'hardening' and testing of the HTTP protocols between clients and server, the authentication service, the instructor and student functions and their interfaces has occurred between January-March 2010. The deployment in the classroom followed then between April-May 2010. We have selected the "Systems Programming Class", cs241 class in Spring 2010 (co-taught by Prof. Kravets and Prof. Godfrey) for the deployment of 80 G2 and 40 G1 phones.

As Figure 2 shows, mi-clicker application suite consists of server side and client side. We will describe few technical details of the application architecture.

mi-clicker Server Side technical details: The server side consists of a SQL database and PHP files. The User Interface that is used by instructors is designed using HTML CSS templates. The SQL database is used to store the authentication information of users, text messages, announcements, quiz and audio files. The token and password that is used for authentication is stored in its encrypted (md5) form in the database. The function performed by the PHP files was that of retrieval of information from the database and sending it to the client. In the case of instructor's web interface, the PHP files not only retrieved and stored information it also displayed statistics using a histogram. The server returns the information in the XML format which is then read by the client using the Document Builder class of the Android platform.
Initially all requests sent to the server were going over non-persistent HTTP connections and there were delays experienced by the user. The code was modified over the summer 2010 to have all requests go over persistent HTTP connections, reducing the setting and tearing down of a HTTP connection each

time a request was made. The server configuration information was checked to ensure that sufficient number of multiple clients can use the mi-clicker application simultaneously.

Mi-clicker's database was designed for extensibility for the whole department. Hence, multiple classes and multiple enrollments for a student are supported. Nevertheless, mi-clicker has strict access control. Only students registered in the cs241 class were able to access the class-specific information and publish class –specific content (voice recording and quiz answer).

mi-clicker Client Side technical details: All activities started by clicking on one of the icons on the screen after the user was authenticated and the class was chosen. A lot of thoughts went into organizing information on phone's small screen, like how to label the already read announcement, how questions should be laid out for both quick browse and examination, and whether the quiz is answered correctly. Built over less powerful phones, in comparison to laptops, we packed all necessary data together for batch communication and cached data on phones during WiFi blackout to avoid data loss. Minimizing the keyboard input in touch screen devices was another concern. We saved users' states in phones to avoid multiple password inputs when users accidentally exited the application. For security, we had preliminary user id, password and token-based authentication.

The *announcement* was implemented as the only activity which used the Background service provided by the Android Operating System. When the user was in the announcement activity, the reminder service which was a thread continually queried the server and updated the announcement list. Hence the user had the latest announcements when he/she signed into the application.

The *Quiz* application retrieved the Quiz list and Quiz file in the XML format which it parsed using the Document Builder class. The Quiz application allowed user to save her answers and submit at a later date. The latest answer was the one stored in the database. The auto-submit feature had a thread running simultaneously which submitted answers as they were selected. The users were not only able to submit their answers they were also able to view the correct answers once they were released by the instructor. The application highlighted the correct answer in green and the incorrect one in red. Threads were used to draw the features on the screen and also simultaneously listen to button click events after.

The Audio and Text Message applications were dialogs and not activities. The recorded text message and audio were sent without the additional overhead of creating separate activities for them.

The Http requests were performed using the *HttpClient* class which allowed for persistent HTTP requests from client to server. The originally used class *HttpURLConnection* allowed only for non-persistent connections with every request a new connection had to be set up and tear down. With this new modification we were able to achieve multiple requests with a single HTTP connection and hence reduce the latency of user requests to the server. We saw a definite performance improvement with this modification.

7. CS 241 Class Deployment

The above sections 3, 4, 5 and 6 showed some answers to the **Question 1**: *how would we (students) want to use phones as educational tools (what kind of educational services do we want on mobile phones)*. The actual deployment of mobile phones G1/G2 in the cs241 class, Spring 2010, provided answers to the **Question 2:** *how would we (department, staff) deploy phones and educational services on a larger scale*, as well as opened up new questions.

Our goal was to deploy the G1/G2 phones to approximately 100 students of the cs241 class, which took place in the classroom 1404 SC. The participation of the students was voluntary since we wanted to test the robustness of the mi-clicker services on larger scale of students, test the WiFi infrastructure in large classroom 1404 SC, test response of a different group of students to the needs of educational tools on phones, and see the complexity of deploying large scale mobile devices.

We had to face the following issues: (a) logistics to distribute, manage and collect G1/G2 phones to/from the students, (b) distribution, setup, configuration, management of updates, usage of mi-clicker educational software, (c) student assistance via help-desk support infrastructure, (e) participation of students.

7.1. Logistics

The first step to the distribution of phones was to create *a lease form* since mobile phones were considered educational devices leased to the students by the department for a specific period of time of their educational process. The prior approaches we have seen in other educational institutions were to create a mobile phone lab, with phones hard-wired to desks, for software development, but not for individual mobile usage. However, since we wanted to explore the mobile usage of phones in educational environments, we wanted the students to carry the phones with them all the time, and use them in and out of classroom. Hence, we have developed processes ***to lease and return a phone*** for the semester duration. The lease form is attached in Appendix 2.

Further logistic issue to be solved was ***short term replacement of phones***. Our process clearly specified when a phone could have been replaced (e.g., hardware faulty – we had memory damage as well as USB plugs faults). The phones were also replaced without charge in case of theft with a proper police report. Otherwise, in case of missing phones, students had to replace them on their own or be charged in the amount of the phone cost.

One important issue was discussed and that is the ***long term replacement of phones***. As we all see, phone hardware is changing almost every 6 months and so the question came up: if we consider phones as educational platform and the educational institution should provide the educational platform (since students pay tuition), can educational institution replace phones every 6-12 months? Where should the funding come from? Within our one year experiment we have seen three generations of Android phones. In August 2009, we have purchased G1 Google phones with Android 1.5 platform, in January 2010 we have purchased G2 Google phones with Android 1.6 and in March-April 2010 time frame Google was distributing Motorola DROID phones with Android 2.0 for educational purposes. So within the short time we had to adjust mi-clicker software to three hardware and OS platforms and these platforms have been supported by short term grants from NSF and industrial sponsors (Vodafone, Qualcomm and Google). However, at this point, it is not clear that the short term grants are the answer towards a long term deployment of mobile phones in classrooms by the educational institutions.

One answer to the long term phone replacement issue could be to build universal educational applications that would run on any mobile platform and so students can actually ***use their own phones*** to download the applets and use them in and out of classrooms. However, under the current diversity of mobile platforms, in terms of hardware, network infrastructure, software, programming languages and their interfaces, it is not clear this solution happens soon.

7.2. Software Distribution, Configuration, Setup, Management, and Usage

We have setup the mi-clicker server on the ***departmental server*** with a secure database that held the names of the cs241 students. The authentication service was setup in such a manner that each student and instructor/teaching assistant had to register through the main MLC mi-clicker website and the system would email the user his/her password. With respect to the mi-clicker application software on the phones, we have used ***Google Android Market*** (applet store) and informed students on the *mi-clicker website* how to download the phone applet. The students could find the instructions at http://mlc.web.cs.illinois.edu/download_page.php . The interface on the Android phone after students downloaded and installed mi-clicker application, i.e., a new icon "*MLClick3r*" (older name for our mi-clicker software) appeared in the slider.

The ***mi-clicker website*** also had information how to connect to our WPA wireless protocol infrastructure, usage instructions for instructors, and student service explanation of the buttons on the phone, and help instructions of error messages.

Based on the students' feedback, we believe that using Google Android Market for phone distribution worked out very well and it also gave us visibility to other users. Students were able to locate mi-clicker app for installation easily and they were notified whenever a new software version became available. However, the important aspect of using Google Android Market was the ***security of the access to the departmental server and student information***, i.e., that other non-Illinois and non-cs241 users, if they downloaded mi-clicker applets on mobile phones, did not get any access through any security hole to our department server(s) and services/information. We have been debugging especially this particular functionality.

7.3. Help-Desk Infrastructure

As with any other software distribution and new deployment, we have setup educational and system support infrastructure for students in case questions occur during the usage of the phones and mi-clicker application. We have setup a "***frequently asked questions***" page on the mi-clicker website, the undergraduate students who developed the mi-clicker software early Spring 2010, setup and rotated ***office hours*** every week to assist cs241 students if they had problems with installation, downloading, operating the phones and the mi-clicker application. Last, but not least, we have established a ***mailing***

list where cs241 students could email any questions regarding the phones and mi-clicker software usage, bug reports, and comments on performance. The undergraduate developers and testers themselves used ***wiki*** to collect bug reports and planned for new updates.

We believe that if mobile phones are going to be deployed as educational tools in the department, ***similar help-desk infrastructure*** will need to be setup to assist for smooth usage of educational tools.

7.4. Students Participation

In the cs241, Spring 2010, we made the student participation voluntary due to the questions about the robustness of the mi-clicker software, accessibility of WiFi infrastructure to many students at the same time and number of phones for the class (we did not have phones for each student of the class). The participation of students was time limited between April 1 and May 15, 2010, i.e., during the second half of cs241 in Spring 2010.

However, in the long term, if phones should serve as stable educational tools, it will be important to differentiate (a) between compulsory and voluntary student participation, (b) between short-term per-semester usage and long-term four years usage of student phone, and (c) between private, public and closed per-class services available to students.

7.5. Deployment Process and Lessons Learned

We have deployed approximately ***40 G1 and G2 free phones*** in cs241 class. The students used the phones in class and also for their own personal usage. The G1/G2 phones allowed them to ***include their own SIM card*** and so many of them put their own SIM cards into the phones and used their own data and voice service. (Note: As a department we are not allowed to pay for data or voice service from governmental funds, hence all our educational services were running over WiFi infrastructure. However, when students used their personal SIM cards, they could access any other services, e.g., Instant Messaging, email, and other services available on Android phones through Verizon service provider).

We have encountered several issues as we deployed phones in the class cs241. In order to advocate phone apps in classroom, phone devices must have essential GUI (Graphical User Interface) support for enterprise WiFi configuration in the future. We expected that students could configure secure campus WiFi access without any trouble and such configuration effort would be one-time task or at least an infrequent task. In Android 1.5 (G1 phones), there was no GUI support for WPA enterprise configuration. Students had to either fall back to web-based authentication over unsecured WiFi, or hack the OS kernel to modify WiFi configuration file. Both approaches were cumbersome. The first approach required repeated manual authentication activities using keyboard before each class. The second required from the students proficiency in Linux command line. Luckily, when we actually deployed phones, Android 1.6 was released with GUI support for enterprise WiFi. We then upgraded all of our G1 phones to Android 1.6.

The students used the mi-clicker phone few times in class for quizzes and announcements. The text messaging was used more often between the students and the teaching assistant. The voice messaging was not used. The students used the phones extensively for their personal usage outside of the classroom. As we will discuss in Sections 8 and 9, many difficulties occurred using mi-clicker software as well as phones within the classroom. The challenges and difficulties can be summarized as follows: (a) difficulty with the wireless infrastructure inside of the classroom when large number of students aim to access resources at the same time, (b) issues with the authentication service, (c) unclear understanding from the instructor and students sides how to include usage of phones into the lecture and educational process, and (d) difficult interfaces to use. The user concerns were extracted from the "Exit" Questionnaire that we asked students of cs241 to fill out when they returned phones. We discuss the results in the Section 8. The infrastructure and resource concerns were extracted from controlled experiments and phone measurements we have conducted in the cs241 classroom and results are reported in Section 9.

8. Students Feedback – 'Exit' Questionnaire

At the end of the cs241/Spring 2010 semester, we have asked students who agreed to participate in the Android/mi-clicker experiment to fill out the "Exit" questionnaire when they return the phones. 19 students responded to our "exit" questions. This 'exit' information helped us in several dimensions to answer the major **Question 1**: *what should be the educational services on phones*, and **Question 3**: *is our departmental infrastructure (staff, instructors, WiFi infrastructure) ready for large scale phone/educational service deployment*. We have asked students questions in three categories: (a) what were students' experience with the mi-clicker software, (b) what kind of problems did students experienced with the phone and software, (c) what kind of future usage of phones/software do the students envision in an educational environment.

The most frequent answers for the first category (experience with mi-clicker) were: (1) ***class did not use mi-clicker often enough***, (2) ***texting with teaching assistant was useful***, (3) ***quiz when used was very useful***, and (4) ***mi-clicker did not integrate well*** with the classroom environment and the cs241 teaching style.

The most frequent answers for the second category (problems with mi-clicker and phones) were:

(1) ***login and password issues*** - students got assigned password and could not change it (and passwords were hard to remember); Furthermore, the password was not cached in the phone, so on every authentication request, students needed to enter the password again.

(2) ***authentication issues*** - it took very long time to get authenticated and even once one got authenticated, due to short in-activity, wireless connections were lost and new re-authentication was needed. This problem proved to be especially problematic in case of quizzes when students at the same time need to authenticate, receive the quiz from the instructor/server and submit answers within a certain time interval.

(3) ***device power issues*** – the G1/G2 devices (especially G1 devices) have very short device life-time. In addition, mi-clicker application was heavy on power usage (no power-optimization in the code), hence often within 1-2 hours phones were out of power. Since the 1404 SC classroom did not have any power-

jacks available to students, re-charging of phones was not possible. Hence, if the students did not come with fully charged phones to the classroom, they may have not be able to take the quiz.

(4) ***WiFi availability issues*** – the wireless infrastructure availability proved to be unstable and access was not always possible. Again for critical educational applications such as taking quizzes inside of classroom this will be a major requirement to stabilize the WiFi infrastructure inside the classroom and make access to all phones, laptops and other mobile devices anywhere in the classroom and anytime.

(5) ***slow access to web-server*** – when students submitted quiz answers or requested a list of announcements or aimed in any other way to communicate with the web-server, the access was very slow which again does not present viable solution for time-critical educational applications such as quizzes, messaging, and other educational tasks.

The most frequent answers to the third category of questions (future potential phone usage) were:

(1) ***teach Android programming in various classes*** – students responded that they see usage of phones in the educational environment as very important and asked to have opportunities to learn program these phones in classes such as operating systems (CS 423), embedded systems (cs431), or new courses on application development.
(2) ***provide simple and stable applications in large classes*** – students would like to see applications such as quizzes, voting, access to class material, and notifications in large classes such as PHYS 211, 212, CS 241, and others.
(3) ***need to access general tools*** – students would like to have access to email, newsgroups, web-browser, ssh and other general tools outside of classroom.

One student especially commented that "I believe that programming experience on a mobile platform is something all CS students should know. "

9. Evaluation of Infrastructure and Software

With the number of G1/G2 phones we had available during Spring 2010, and DROID phones in Summer 2010, we have conducted small scale and large scale experiments to evaluate the performance of the software mi-clicker and the wireless infrastructure. We wanted to answer the **Question 3**: *is our departmental infrastructure ready for large scale deployment of phones in and out of classroom*.

In this section we will investigate the readiness of our communication infrastructure and consider two major aspects: (a) **what are the HTTP request quality measurements under the WPA Enterprise networks** (since our mi-clicker application tool-kit is web-based and concerns have been expressed about the slow access to the web-server), and (b) **what are the WiFi conditions in the classroom** 1404 SC (since most of the student concerns included WiFi availability in the 1404 SC classroom). The first aspect will be measured with respect to the ***success ratio*** of HTTP requests to the web-server, ***RTT (Round-Trip-Time)*** and ***available bandwidth***. The second aspect will be measured under *different scale of phones, different topologies, different WiFi authentication protocols, different power provision, with heterogeneous devices, and with goals of finding dead spots*.

9.1. Large Scale Android Phone Experiments in Spring 2010

Experimental Environment: We have conducted large-scale Android phone experiment during the Spring Break of 2010 in the Siebel Center, room 1404 which has 200 seats. The reason for the time frame was that the classroom was available to us through the whole week and so we could conduct multiple and diverse experiments. The numbers of phones in the experiments were 80+ G2 phones and 40+ G1 phones. We have conducted two types experiments: (1) **large scale experiments with phones only (Pure Large Scale)**, and (2) **large scale experiments with mixture of phones and laptops (Mixture Large Scale)**. In the first experiment type we have included 120+ phones and in the second experiment type we have included 20 or 80 G2 phones plus several laptops.

Scenarios: We have considered four scenarios of the phones and laptops.

First scenario represented the '***Pure Large Scale'*** experimental setup with **random placement of 80+ G2 and 40+ G1 phones** over ***WPA2*** infrastructure within the 1404 SC room area with anticipated access points (Note: We have anticipated four access points (APs) in the scenarios and measurements, but as it turned out after our measurements, there were only two APs available) as shown in Figure 10-left. Phones were randomly assigned either TTLS or PEAP for WPA2 enterprise authentication. The experiment in this scenario lasted 1 hour with phone being continuously charged.

Second scenario belonged to the '***Mixture Large Scale***' experimental setup with **random placement of phones** within the 1404 SC room as shown in Figure 10-right. Phones were randomly assigned either TTLS or PEAP for ***WPA2*** enterprise authentication. The experiment in this scenario lasted 1 hour with phone being continuously charged.

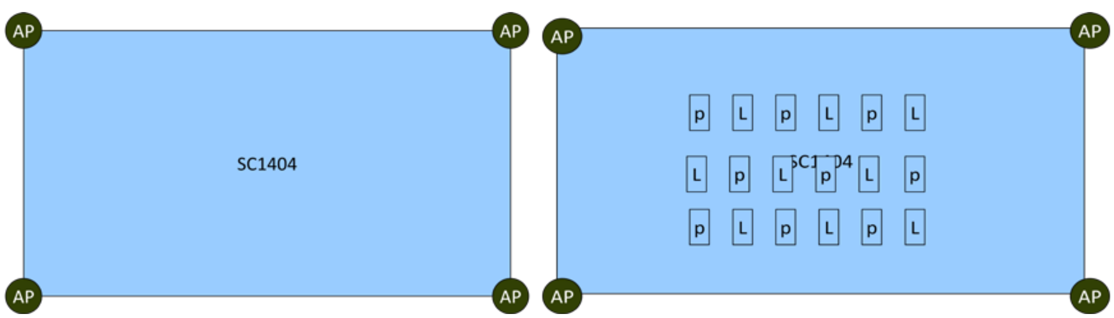

Figure 10: *Pure Large Scale Experiment with (left) random placement and Mixture Large Scale with random placement (right).*

Third scenario belonged to the '***Pure Large Scale***' experimental setup (Figure 10-left) with random placement of 20 G2 phones at the last few rows in the classroom without power plug. Phones used ***UIUCnet*** unencrypted WiFi communication. The experiments lasted 1 hour.

The *fourth scenario* was similar to the second one and belonged to the '***Mixture Large Scale'*** experimental setup (see Figure 10-right), except that phones used browser-based authentication and messages were not encrypted over ***UIUCnet***. The experiments lasted 1 hour .

9.1.1. HTTP Request Quality Results

To measure the WiFi and HTTP performance, we have prepared a testing script, running on each phone in 1404 SC classroom using above specified scenarios.

Experiment Description: In this experiment we were running test scripts between Android phones and a simple web-server (NO mi-clicker software) to stress-test the HTTP protocols and the access to the web-server over WPA enterprise network. We have tested the same HTTP test-scripts in the 1404 SC classroom as well as at home, and we tested the HTTP requests on phones and on laptops.

Sketch of testing script: We have generated a random http request, selected from two categories; and repeated until the end of testing period.

(a) To a MLC server with probability 0.6: operations are packed in a session containing the following steps;
   (1) Authenticate users by username, password, retrieving a token for future communication in this session; pause 1-3 seconds
   (2) 10-15 steps randomly selected from following operations, with 2-5 seconds pause between successive steps ([x] means with probability x)
      a. Get the list of all enrolled class [0.15]
      b. Get the list of quizzes for a class [0.15]
      c. Get questions of a quiz and upload answers [0.2]
      d. Upload text messages [0.2]
      e. Upload voice message [0.1]
      f. Get announcement for a class [0.2]
   (3) Log out.
(b) To popular websites with probability 0.4: randomly selected from amazon.com, CNN.com, times.com, bing.com, with approximately 100,000 bytes/page

Overall Results Summary: The experimental results showed a long **RTT delay for a single HTTP request**. At Siebel Center, the RTT delay was **4-8 seconds** for a phone HTTP request versus less than ***1 second*** in a home environment. At Siebel Center, the RTT delay was 4-8 seconds for a phone HTTP request to the web-server versus less than 1 second for a laptop. Hence, the RTT delays clearly indicated that the HTTP request delays were unacceptable for interactive applications such as mi-clicker and changes need to be considered.

Another metric we have measured and that was the **failure ratio of HTTP requests** to the web-server. Our results showed that HTTP requests on phones showed very high failure ratio up to 37-64%, which was again not acceptable for interactive applications such as quizzes in mi-clicker. Table 2 summarizes the results.

Metrics/Results	Siebel Center/Phones	Home/Phones	Siebel Center/Laptop
RTT (Round Trip Time) of single HTTP request	4-8 Seconds	< 1 second	< 1 second
Failure Ratio of HTTP requests	37-64%	-	-

Table 2: *Results of HTTP Requests to Web-Server over WPA Enterprise Network*

9.1.2. Detailed WiFi-Related Results

Based on our overall Spring 2010 measurements, we have concluded that (a) **WiFi stability was low** in the 1404 SC classroom, and WiFi (re-) association/handoff was a big issue; (b) **G1 phones performed worse in terms of WiFi connectivity than G2 phones**. Hence, after this finding, we have done most of the WiFi experiments with 80+ G2 phones; (c) **WPA Enterprise networks had longer delays than UIUCnet** due to authentication, encryption, decryption overheads in the range of 0.2-0.6 seconds. UIUCnet does not have encryption; (d) **SC 1404 classroom had insufficient deployment of access points (APs)**. The anticipation and information from the system administration was to have four APs for the SC 1404 classroom, but after our measurements, we found only two APs served this large classroom. We have found dead spots in the middle of the classroom (very low SNR signal), and with large number of phones, we experienced congestion which increased the end-to-end delay dramatically, with RTT up to 4-6 seconds per HTTP request.

Below we present in detail our measurements of the WiFi-related infrastructure results.

Experiment 1: In this experiment we had deployed maximum number of phones 80+ G2 phones and 40+ G1 phones in SC 1404 classroom with random phone placement and using WPA2 enterprise network for 1hour. We have also done random assignment for TTLS (Tunneled Transport Layer Security)/PEAP (Protected Extensive Authentication Protocol). All phones have been connected to power plug (we have brought into the classroom power extension courts to connect all phones to power jacks which is not available during regular classroom times.). The ***Scenario 1*** setting was as shown in Figure 10 (left).

Under this experimental setup, we had 26 hosts respond with the HTTP request. The failure ratio was around 64% and delays to access the web-server were approximately 4.5 seconds to a regular website and 6 seconds to the MLC mi-clicker website. Results are shown in Table 3.

In this experiment, even though 120 phones were deployed, only 26 phones reported their results back to the database during 1 hour period. This was due to the fact that we had to manually start the Mi-Clicker applications in every phone, ensuring WiFi is connected when the app starts. The manual starting process for all the phones took more than half an hour.

	TTLS	PEAP
Fail Ratio	0.6527	0.6349
Delay (web) in seconds	4.4940	4.5240
Delay (mlc) in seconds	5.8872	6.1212

Table 3: *Experimental Results under Scenario 1/Experiment 1*.

Experiment 2: In this experiment we have removed the G1 phones due to their low WiFi performance, and considered 80+ G2 phones with laptop interference (***Scenario 2***). We have considered random placement of phones and laptops were adjacent to phones with streaming video to stress the interference effect. We have used the WPA2 enterprise network for 1 hour. We have employed random assignment for TTLS/PEAP.

Under this experimental setup, we had 52 nodes respond, the failure ratio of HTTP requests was approximately 50% and the delay to a regular website was approximately 4.1 seconds and to the MLC mi-clicker web-server approximately 5.47 seconds. Results are shown in Table 4.

	TTLS	PEAP
Fail Ratio	0.4524	0.5667
Delay (web) in seconds	4.1727	4.0887
Delay (mlc) in seconds	5.4338	5.5254

Table 4: *Experimental Results under Scenario 2/Experiment 2*.

Experiment 3: The HTTP request times to web-servers have been large in Experiment 1 and 2, and hence we wanted to test how a HTTP request will perform under ***UIUCnet*** network. Hence, we have deployed 20 G2 phones with random placement of phones according to ***Scenario 3*** (**Figure 10-left**). We have tested the phones under ***UIUCnet*** for 1 hour and the phones have been without power plug. In this case, on average 9 hosts responded out of 20, hence we achieved failure ratio of 0.3748 with delays of 1.8724 seconds HTTP request to a regular website and 2.3630 seconds HTTP request to the MLC mi-clicker website.

Experiment 4: We have altered the Experiment 3 with adding more phones and laptops according to the ***Scenario 4*** (Figure 10-right). We have experimented with 80+ G2 phones, random placement of phones and placed adjacent to them laptops for interference. The HTTP requests were running over UIUCnet networks for 1 hour. At this point, on average 40 hosts responded, yielding failure ratio of 0.4843 and delays of 3.5941 seconds for regular website and 5.3375 seconds for MLC mi-clicker website.

9.1.3. Possible Reasons for Long Delays

As we have discussed above, having 4-8 seconds access to the MLC web server is unacceptable for the v interactive educational applications such as mi-clicker. Hence, we have analyzed some of the possible reasons for the long delays to take next steps for future generations of MLC educational tools.

We have found several problems with the software and the phone themselves (without laptops):

- **mi-clicker implementation** – mi-clicker used non-persistent HTTP connections have been used, contributing on every HTTP request to additional delays due to re-association of connections.
- **Login and password** – mi-clicker used fixed assigned passwords which were difficult to remember and users were not able to change the passwords. Also, passwords could not be cached, so on every loss of WiFi and/or server connection, password had to be retyped, causing additional delays.
- **Authentication process** – as experiments showed, using WPA2 enterprise networks infrastructure induced additional authentication and security overheads. Phones were equipped with small memory and less powerful CPU, thus adding visible delay overhead in end-to-end delay.
- **Congestion** – as experiments showed, we had only two access points available, hence congestion and bandwidth availability represented difficulties when deploying 100+ phones.
- **Wrong setup of phones** – when we were running test scripts, phone screens timed out which caused shut down of the WiFi connection and then again WiFi re-association.

In addition as we added laptops into the mixture (Scenario 2 and 4), which is an expected setup in a classroom since students will bring with them not only phones but also their laptops, we saw additional problems:

- **Laptop interference** – since laptops were placed adjacent to phones, laptops transmitted with larger power than cell-phones, preempting the cell-phone transmission for several rounds, causing phones additional delays with HTTP requests. This is a problem that represents one aspect of the well-known hidden terminal problem.
- **AP preferred treatment of laptops** – access points gave better chance to laptops to decode laptops' packets than phones' packets when they sent packets simultaneously.
- **Phones reactions to interference** - when phones saw packet drop, their protocol backed off, causing unfairness in the access to the wireless channel between phone and laptop.

Another cause of the large web access delay may have been that the phones' wifi interfaces adopted PSM (Power Save Mode) instead of CAM (Constantly Awake Mode). The 802.11 standard supports multiple modes of operation, with different power consumption for each mode. When CAM is adopted, the WiFi device stays awake, ready to send and receive packets all the time, hence consumes most energy. On the other hand, when PSM (Power Save Mode) is adopted, the WiFi device goes to sleep, i.e.

it stops receiving packets when it has no packet to transmit, and wakes up periodically for beacons. If the beacon indicates there are packets for it buffering at the AP (access point), the wifi device sends a separate PS-POLL message to receive each buffered packet. The 802.11 Access Point (AP) supports PSM by 1) buffering incoming packets for WiFi clients in PSM, 2) indicating the presence of buffered packets via fields in beacon messages, and 3) delivering the buffered packets after the client notifies the AP it's ready to receive one or more packets.

While the static PSM approach allows the device to save power by only waking up when packets are outstanding at the AP, it introduce extra latency involved in receiving packets via PS-POLL and such delay has been found to be high for interactive applications such as web browsing [KrBa2002]. The APs in the WLANs of UIUC set default beacon interval value to be 100ms, therefore each packet delivered to the phone from the AP has extra delay caused by buffering at the AP which is up to 100ms. Since HTTP connection takes several rounds of handshake to build, the phones may take 400-500ms to build a HTTP connection. In order to mitigate this problem, many WiFi devices today also implement a technique known as ***adaptive PSM***, where the device switches between PSM and CAM based on some heuristics.

According to our observation, the G1, G2, and droid phones that we used in the experiments all adopted adaptive PSM, and the switch between PSM and CAM was triggered by reception of a threshold number (denoted as *TH*) of packets or lack of network activity for a pre-defined duration. The device notified the AP of its transitioning to PSM or CAM by sending NULL data frames with the power management bit (in the MAC header) set to 1 or 0, respectively. However, we found out that the default value of *TH* on the G1, G2, and droid phones was too large, so that the traffic flow of our python program was not large enough to trigger the phone operating at CAM mode. Therefore in our experiments, the phones still encountered large delays caused by PSM, while the laptops which always operate at CAM mode worked better.

We reset the *TH* on the droid phones to a smaller value and observed significant delay decrease, while at the same time, the power consumption also increased. However, we found out that this solution did not work on the G1/G2 phones' WiFi interfaces. When we reset the *TH* to a G1/G2 phone smaller value, although the phone actually operated at CAM mode when the traffic flow was large (we observed so using a WiFi sniffer), it set the power management bit (in the MAC header) to 1 (indicating PSM mode) in EVERY single packet it sent out. Therefore, the AP assumed that it operated at PSM mode and buffered all packets. We suspect that the discrepancy between the power management bit value and the actual operation mod was a bug in the NIC driver of G1/G2 phones.

Sometimes the software tools we used to build our program could cause the delay. For example, our test program was running using ASE, when we upgraded ASE to a newer version, the HTTP request delay decreased by 40ms.

Besides problems with the phone software mi-clicker implementation, phone configuration, laptop interference, we anticipate that the delays we saw with HTTP requests between phones and web-servers might be also due to using the Meru network devices:

- **Centralized Scheduling** - The Meru networks use centralized packet scheduling algorithms over the whole WLAN. Hence, this can influence the airtime fairness, seamless handoff, and overall scheduling time (*Note: We do not have any solid proof that Meru scheduling algorithm introduces the extra delay. We only implicitly observed that sometimes the AP stopped serving our devices (both laptops and cell-phones) for over 500 ms or even 1 second. Since all those devices were not moving during the measurements, we could only suspect that the APs were waiting for new instructions about the transmission schedule.*)
- **Laptop Access Times** – We have done laptop measurements in the Siebel Center (in promiscuous mode) and it happened frequently that our laptop received 0 packets from the access point for over 2 seconds. This is again very undesirable if we consider interactive learning applications in classrooms. We suspect that the Meru AP stopped serving clients when it processed handoff or other scheduling, and this processing time could be as long as several seconds.

9.2. Small Scale Phone Experiments in Summer 2010

In Summer 2010, we have taken the feedback from Spring 2010 and made several changes to the mi-clicker software to speed up the RTT of HTTP requests to the MLC web-server. We have experimented with small number of DROID phones.

Changes: We have introduced persistent HTTP connections within mi-clicker in order to improve the network performance and reduce delays. The original design had each HTTP request open a connection and tear it down. This introduced delays in the user's interaction with the mi-clicker application. The networking code was modified during summer to convert all the non-persistent requests to persistent HTTP requests. Further optimization like disabling Nagle's algorithm at the cost of lower bandwidth was done by enabling the TCP_NODELAY option in the *HttpClient* class of the Android OS.

Experimental Description: Three DROID phones were used in 3 different locations in the Siebel Center building. The locations were where students would most likely be present during the course of the semester. The classroom, coffee shop and the computer lab were used to test small scale usage of the mi-clicker application. The mi-clicker experiment was only tested over the Wi-Fi infrastructure since the aim was to find the connection which provides the minimum delay for the user requests. Both the non-persistent HTTP connection and persistent HTTP connection were tested in all three locations with queries being made for 15 minutes. The time was chosen to be the maximum amount of time that a user would use the phone in class to take a quiz, check announcements and ask a question. Since we were looking into persistent connections, we had to make sure the connection stayed open for a sufficiently long amount of time so that re-authentication could be avoided. The server specified that the timeout was 5 minutes for all client requests. The maximum users that could connect to the server was 150.

Some preliminary measurement results of the above experimental setup are shown in Table 5.

Activity	Non-persistent Http connection (in ms)	Persistent Http connection (in ms)
Authentication of User	382.8	398.7
Retrieving class lists	203.9	152.7
Sending a text message	341.8	300.2
Retrieving quiz lists	268.6	156.3

Table 5: *Experimental Results of mi-clicker HTTP requests using DROID Phones*

The results clearly show improvement of the access time when compared individual operations between non-persistent and persistent http connections. Further measurements are and will be going on in Fall 2010 and Spring 2011 in a larger scale experiment to compare the results between the implementation in Spring 2010 and Summer 2010. The mi-clicker software will be also further improved by adding additional features and optimizing it for a more user friendly experience.

10. Related Work

Advances in wireless technologies have allowed users to communicate ubiquitously anywhere and anytime and made it possible for users to access and exchange information through wireless handheld devices such as cell phones, PDAs, tablets, and other wireless devices [LyYo2002]. We are seeing an explosion of wireless communities and applications (e.g., Buddy Finder, Proximity Dating, and Cab Ordering) as wireless infrastructures and deployments in wireless digital cities and wireless local communities emerge pervasively. For example, in Helsinki, Finland, people discuss, plan, and manage local events with cell phones through the Helsinki Virtual Village [Helsinki2008]. On the other hand, wireless local communities are fixed to a specific area, e.g., a shopping center or tourist park [Sun2007], to share their local experiences with each other.

It is important to stress that these wireless communities are very different from traditional Internet-based online communities, in which people are connected with one another but neither know each other in person nor necessarily care where they are at any specific moment. More specifically, although people in different places are connected through Internet-based communities, information exchange alone is not likely to lead to a sense of social interaction close to what happens in the real-world communities [SpPa2004]. In real-world communities, social interactions occur in physical and social contexts that are shared by those involved. However, most contextual cues are filtered out in text-based communication through the Internet. That is why face-to-face communication has usually been found to be more effective than computer-mediated communications at fostering community [Whittaker 2003].

In contrast, in wireless communities, people are much more closely bound to each other, through a sense of sharing a common physical and/or social context. In such communities, for example, it is possible for members to access the contextual cues directly or with the help of information technologies. For mediated communications, research has found that joint attention and social linkage are necessary conditions for effective information exchange [ClMar1981; NaWhi2002]. As a result, a sense of sharing the same physical and/or social context helps to bind people more closely in wireless communities, leading to “contextual communality.”

The realization of *mobile learning communities* will also depend on prior work in the space of (1) active participation of members [WhiIsaO1997], (2) the design and development of advanced mobile application services for wireless communities/users that will facilitate the sharing of contextual cues, (3) assurance of stable underlying software and hardware infrastructure (i.e., reliability, robustness, privacy, and security of services in the mobile educational environments), and (4) assurance of social trust from the various users.

11. Final Conclusions and Lessons Learned

Over the one year, we have explored various dimensions of mobile learning communities in our department at the University of Illinois with some feedback from students at TU Darmstadt. The basic hypothesis is that our universities will have students who carry phones (not only laptops) and they will create mobile learning communities during their time at our campuses. So we need to think about if we as formal educational institutions are ready for these mobile learning communities. The one year study aimed to answer the three questions: (1) what kind of educational tools are appropriate for mobile phones within mobile learning communities, (2) how do we deploy the software/hardware phones in large scale classes, and (3) is our infrastructure ready for this large scale deployment. The answers to all of the questions are complex and include large number of considerations before we respond with the final answer "Yes, we are ready for phone-based mobile learning communities". As we showed, some aspects of the MLC educational process can be mastered within short time period such as establishing logistics of the phone deployment, but some aspects of the MLC educational process will take several iterations to provide stable MLC environment for our students and instructors. So at this point, our answer is "***We are not quite yet there*.**"

For our own efforts the next steps will be to:

(a) continue with efficient mi-clicker protocol implementation, enhancements with functions for usage outside of classrooms and usage inside of classroom with smaller discussion groups, and continued deployment in classes;

(b) implement other/new educational applications for phones as student surveys suggested,

(c) continue with detailed analysis of our communication infrastructure by keeping track of statistics for each client such as TCP dumps, RSSI, SNR at the PHY layer, and retry indicator at the MAC layer,

(d) investigate Meru WiFi network infrastructure via Meru software logs of AP association, re-association and hand-off.

Besides the various software improvement efforts, we plan to continue working with CITES campus infrastructure group to improve WiFi (Meru network) infrastructure in and out of classrooms, to introduce more AP nodes around large classrooms, to tune some control parameters of the AP so that the scheduling overhead won't affect the performance the clients even when the number of clients is large, and to consider more power plugs in classrooms. Besides improvements of our communication and computing infrastructure, we need to enhance on teaching mobile phone programming and introduce for example teaching Android programming in various of our undergraduate classes. With this effort, we will increase the interest of our students and instructors in mobile platforms as educational tools and increase the number of applets built for our own department.

In summary, we believe that the deployment of phones as educational tools in and out of classrooms should continue. As we experiment with phones and other mobile devices, and increase the number of students who can program new educational and other useful applications, the mobile learning community at Illinois becomes stronger, giving strong advantage to our students in their future.

Acknowledgement

This effort has been funded by the National Science Foundation (NSF) CNS IT grant NSF 0855129, and industrial sponsors such as Vodafone Educational Grant, Qualcomm and Google. The opinions presented in this document are our own and do not represent the opinion of the funding agencies and sponsors.

We would like to thank the mi-clicker undergraduate developers: Alex Loeb, Michael Ford, Craig McIntyre, Ohm Patel, Samuel Retika, Ang Pan, and the mi-clicker testers: Ben Gass, Wonjun Jang, Min Koo Kang, Yudha Purnama Malimas, Andrew Rachwalski. We would also like to thank students of the class cs241 and cs425 at UIUC, and students in the class Comm I, taught by Prof. Steinmetz, at TU Darmstadt, for their cooperation in the MLC project.

References

[ACI] NSF Partners With Google and IBM to Enhance Academic Research Opportunities, NSF Press Release 08-031.

[AWS] Amazon Web Services. http://aws.amazon.com

[Adve2002] S.V. Adve, A.F. Harris, C.J. Hughes, D.L. Jones, R.H. Kravets, K. Nahrstedt, D.G. Sachs, R. Sasanka, J. Srinivasan, and W. Yuan. "The Illinois GRACE Project: Global Resource Adaptation through CoopEration," in the *Proceedings of the Workshop on Self-Healing, Adaptive, and self-MANaged Systems (SHAMAN)*, 2002.

[BK2001] J. Barton, T. Kindberg, "The Cooltown user experience," HPL-2001-22. 2001.

[Bulusu2000] N. Bulusu, J. Heidemann, and D. Estrin. "GPS-less Low-Cost Outdoor Localization for Very Small Devices," *IEEE Personal Communications Magazine*, Vol. 7, No. 5, October 2000, pp. 28-34.

[CaKr2002] C. Carter and R. Kravets. "User Devices Cooperating to Support Resource Aggregation," in the *Proceedings of WMCSA*, 2002.

[CaKrTo2003] C. Carter, R. Kravets, and J. Tourrilhes. "Contact Networking: A Localized Mobility System," in the *Proceedings of IEEE Mobisys*, 2003.

[CaRaKr2003] C. Carter, P. Ratanchandani, and R. Kravets. "Manycast: Exploring the Space between Anycast and Multicast in Ad Hoc Networks," in the *Proceedings of Mobicom*, 2003.

[CCT] HP Labs cloud-computing test bed: Technical overview. http://www.hpl.hp.com/open_innovation/cloud_collaboration/cloud_technical_overview.html

[Cheema2005] A. S. Cheema, M. Muhammad, and I. Gupta. "Peer-to-peer discovery of computational resources for Grid applications." In Proc. IEEE/ACM Workshop on Grid Computing (GRID), 2005.

[ClMar1981] H. Clark and C. Marshall. "Definite Reference and Mutual Knowledge," in A. Joshi, B. Webber, I. Sag (Eds.), *Elements of Discourse Understanding*, Cambridge University Press, 1981.

[Crep2007] R. Crepaldi, A.F. Harris III, R. Kooper, R. Kravets, G. Maselli, C. Petrioli, and M. Zorzi. "Managing Heterogeneous Sensors and Actuators in Ubiquitous Computing Environments," in the *Proceedings of the First ACM Workshop on Sensor Actor Networks*, 2007.

[Davis2007] Francis M. David, Bill Donkervoet, Jeffrey C. Carlyle, Ellick M. Chan, Roy H. Campbell. Supporting Adaptive Application Mobility. In 2nd International Workshop on Pervasive Systems (PerSys 2007). November 2007.

[GAE] Google App Engine. http://code.google.com/appengine

[GoG] GoGrid. http://gogrid.com ; http://www.nsf.gov/news/news_summ.jsp?cntn_id=111186

[Gra1983] M. Granovette, "The Strength of Weak Ties: A Network Theory Revisited," Sociological Theory, Vol. 1, 201-233, 1983.

[GupBir2003] I. Gupta, K. P. Birman, P. Linga, A. J. Demers, and R. van Renesse. "Kelips: building an efficient and stable P2P DHT through increased memory and background overhead." In Proc. 2nd International Workshop on Peer-to-Peer Systems (IPTPS), 2003.

[HaKr2003] A. F. Harris III and R. Kravets. "Pincher: A Power-saving Wireless Communication Protocol." Technical Report UIUCDCS-R-2003-2368, University of Illinois at Urbana-Champaign, 2003.

[HaSnKr2007] A. Harris, R. Snader, and R. Kravets. "Energy-efficient Transmission for Multimedia Streams in Last-hop Wireless Internet, in *Proceedings of IEEE Broadnets*, 2007.

[Helsinki2008] www.arabianranta.fi.

[InkScribe2008] www.inkscribe.com.

[Jin2006] G-Y. Jin, X-Y. Lu, and M-S. Park. "An Indoor Localization Mechanism Using Active RFID Tag," in *Proceedings of IEEE SUTC (International Conference on Sensor Networks, Ubiquitous and Trustworthy Computing)*, Vol. 1, 2006, pp. 40-43.

[Kapa2007] A. Kapadia, P. Naldurg, and R. H. Campbell, "Distributed Enforcement of Unlinkability Policies: Looking Beyond the Chinese Wall," in Proceedings of the Eighth IEEE Workshop on Policies for Distributed Systems and Networks (POLICY'07), Bologna, Italy, 2007.

[KrBa2002] R. Krashinsky and H. Balakrishnan. Minimizing Energy for Wireless Web Access with Bounded Slowdown. In MobiCom, 2002

[KrCaMa2001] R. Kravets, C. Carter, and L. Magalhaes. "A Cooperative Approach to User Mobility," *ACM Computer Communication Review*, Vol. 31, No. 5, 2001.

[Lad2004] A.M. Ladd, K.E. Bekris, A.P. Rudys, D.S. Wallach, and L.E. Kavraki. "On the Feasibility of Using Wireless Ethernet for Indoor Localization," *IEEE Transactions on Robotics and Automation*, Vol. 20, No. 3, pp. 555-559, June 2004.

[Liang2007] Jin Liang, Indranil Gupta, Klara Nahrstedt, "Reliable On-Demand Management Operations for Large-scale Distributed Applications," ACM SIGOPS Operating Systems Review (OSR) Special Issue on Gossip-Based Networking, vol. 41, no. 5, pp. 82-88, October, 2007.

[Lim2006] H. Lim, L-C. Kung, J.C. Hou, and H. Luo. "Zero-Configuration, Robust Indoor Localization: Theory and Experimentation," in *Proceedings of the 25th IEEE International Conference on Computer Communications (INFOCOM)*, 2006, pp. 1-12.

[Lind2007] D. Lindquist, T. Denning, M. Kelly, R. Malani, W. G. Griswold, and B. Simon, "Exploring the Potential of Mobile Phones for Active Learning in the Classroom," in *SIGCSE '07: Proceedings of the 38th SIGCSE Technical Symposium on Computer Science Education*, March 2007.

[Linge2004] K. Lingemann, H. Surmann, A. Nüchter, and J. Hertzberg. "Indoor and Outdoor Localization for Fast Mobile Robots," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2004, pp. 2185-2190.

[Lore2004] Y. Chen, X.Y. Chen, F.Y. Rao, X.L. Yu, Y. Li, D. Liu, "LORE: an infrastructure to support location-aware services," IBM Research Journal, vol. 48. no. 5/6, 2004.

[LyYo2002] K. Lyttinen and Y. Yoo, "Issues and Challenges in Ubiquitous Computing," *Communications of the ACM*, Vol. 45, No. 12, 63-65, 2002.

[MaDaScho1995] R.C. Mayer, J.H. Davis, and F.D. Schoorman. "An Integrative Model of Organizational Trust," *Academy of Management Review*, Vol. 20, 709-734, 1995.

[Meier2009] R. Meier, "Professional Android Development", 2009

[McCuChe1998] D.H. McKnight, L.L. Cummings, and N. Chervany. "Initial Trust Formation in New Organizational Relationships," *Academy of Management Review*, Vol. 23, 473-490, 1998.

[Miller2006] L.E. Miller et al., "RFID-Assisted Indoor Localization and Communication for First Responders," in the *Proceedings of ISART*, 2006.

[Molnar2006] I. Molnar-Szakacs and Katie Overy, "Music and Mirro Neurons: From Motion to 'e' Motion", Social Cognitive and Affective Neuroscience 2006 1(3): 235-241

[Murphy2004] A. Murphy, "The 'Strong Tie' Between Social Networks, Job Attainment and Upward Mobility", Annual Meeting of American Sociological Association, San Francisco, 2004.

[NaWhi2002] B. Nardi and S. Whittaker, “The Place of Face-to-face Communication in Distributed Work,” in P. Hinds and S. Kiesler (Eds.), *Distributed Work*, pp. 83-112, Cambridge, MA: MIT Press, 2002.

[NeHaKr2007] S. Nelson, A. Harris, and R. Kravets. “Event-driven Role-based Mobility in Disaster Recovery Networks,” in the *Proceedings of CHANTS*, 2007.

[Pe1992] R. Perkins, “Teaching teachers how to use the computer as a tool and tutor in the classroom”, EERA Conference, Hilton Head, SC, March 7, 1992

[Pong2007] T. Pongthawornkamol, K. Nahrstedt, and G. Wang. “The Analysis of Publish/Subscribe Systems over Mobile Wireless Ad Hoc Networks,” in the *Proceedings of Mobiquitous*, 2007.

[RaKr2003] P. Ratanchandani and R. Kravets. “A Hybrid Approach for Internet Connectivity for Mobile Ad Hoc Networks,” in the *Proceedings of WCNC*, 2003.

[Ravi2006] N. Ravi, P. Shankar, A. Frankel, A. Algammal, and L. Iftode. “Indoor Localization Using Camera Phones,” in the *Proceedings of the 7th IEEE Workshop on Mobile Computing Systems & Applications: Supplement (WMCSA)*, 2006, pp. 1-7.

[SeKr2005] C. Sengul and R. Kravets. “Conserving Energy with On-demand Topology Management,” in the *Proceedings of the IEEE International Conference on Mobile Adhoc and Sensor Systems Conference*, pp. 449-458, 2005.

[SpPa2004] L. Sprout and J.F. Patterson. “Information Cities: Making Information Cities Livable,” *Communications of the ACM*, Vol. 47, No. 2, pp. 33-37.

[Sun2007] J. Sun. “Wireless Local Community in Mobile Commerce,” in A. Tatnall (Ed.), *Encyclopedia of Portal Technologies and Applications*, pp. 1204-1209, Hershey, PA: Idea Group Reference, 2007.

[SuPo2008a] J. Sun, M.S. Poole, and K. Koong. “Critical Success Factors for Context-aware Mobile Communication Systems,” *International Journal of Mobile Communication* (in press).

[SuPo2008b] J. Sun and M.S. Poole. “Beyond Connection: Situated Wireless Communities,” *Communications of the ACM* (in press).

[VoWo1996] M. Vo and C. Wood. “Building an Application Framework for Speech and Pen Input Integration in Multi-modal Learning Interfaces,” in the *Proceedings of ICASSP*, 1996.

[Vu2008] L. Vu, K. Nahrstedt, and M. Hollick. “Exploiting Schelling Behavior for Improving Data Accessibility in Mobile Peer-to-Peer Networks,” in the *Proceedings of Mobiquitous*, 2008.

[VuNa2007] L. Vu and K. Nahrstedt. “Adaptive Mobility-Assisted Data Dissemination In Mobile Disaster/Recovery Environments,” in the *Proceedings of the Military Communications Conference*, 2007.

[Ward2008] J.R. Ward. “History of Pen Computing, Annotated Bibliography in On-line Character Recognition and Pen Computing,” http://users.erols.com/rwservices/pens/biblio05.html, 2008.

[Watt2006] Z. Chambers, A. Chidanandan, R. DeVasher, L. Merkle, M. Minster, S. Mitra-Kirtley, D. Mutchler, S. Sexton, A. Watt, J. Williams, and M. Zoetewey. “What Is Beyond the Laptop Initiative? Perhaps: Tablet PCs and DyKnow Vision Software,” *in the Proceedings of the 36th ASEE/IEEE Conference on Frontiers in Education*, San Diego, CA, October 2006.

[WhiIsaO1997] S. Whittaker, E. Isaacs, and V. O’Day, “Widening the Net: Workshop Report on the Theory and Practice of Physical and Network Communities,” *SIFHCI Bulletin*, Vol. 29, No. 3, 1997, New York: ACM Press.

[Whittaker2003] S. Whittaker. "Theories and Methods in Mediated Communication," in A.C. Graesser, M.A. Gernsbacher, and S.R. Goldman (Eds)., *Handbook of Discourse Processes*, pp. 243-286, Mahwah, N.J.: Lawrence Erbaum, 2003.

[WWW1] http://developer.android.com/index.html

[WWW2] http://www.anddev.org/

[WWW3] http://www.droidforums.net/

[WWW4] http://mobiforge.com/

Appendix 1

Questionnaire for Students in Class 'CS 241' Fall 2009

In Spring semester 2010 we would like to engage in an experiment to use mobile cell-phones such as Android/G-Phones in an undergraduate class for educational purposes. The goal of this questionnaire is to find out what educational applications would **YOU (students)** prefer to have on your cellular phone that would help you in your educational process. **The questionnaire is anonymous (please, DO NOT put your name/UID/email on the sheet of paper). The questionnaire is voluntary.**

1. What usage of a cell-phone do you see inside/outside of a classroom that would help you understand better the material presented by the instructor?
2. What educational applications would you envision on your cell-phone to help you communicate with the teaching assistant of your class?
3. What applications would you like to have on your cell-phone to help you communicate with classmates, TAs, instructor outside of the classroom?
4. What educational applications would you like to have on your cell-phone to help you access class-related information inside and outside of classroom?
5. In some classes you may do projects with other members of your class. What applications would be useful to you in conducting projects with your classmates?
6. Are there any other educational applications that you wish you had on a mobile phone?

Appendix 2

EQUIPMENT LOAN TO EMPLOYEES AND STUDENTS
Authorization for Temporary Off-Campus Use
of University Equipment

Loaning Department: ____Computer Science_______ Chart/Organization Code: _1-434__________

Borrower Name (Last, First): ______________________________________ ____ ______

Check one:

Faculty

Campus Address: ____

Campus Phone: ____

Permanent Address: ____

Permanent Phone: ____

Off-campus location/address of loaned equipment ________________N/A___________________________

Loan Term (may not be greater than one year): From ________________________________ to ________

(Loaning department may request return of equipment before the end of loan term if needed.)

Purpose of Loan:

____

____

Special Conditions:

Borrower is responsible for all voice, data, text message and any other charges not directly related to the CS class or project. Borrower is financially responsible for the equipment and all accessories based on University policy referenced below.

Equipment Data:

Description	Property Control Number	Serial Number
1. ________________________	____________________	____________________________
2. ________________________	____________________	____________________________
3. ________________________	____________________	____________________________

4. ________________________ ______________________ ____________________________

Signatures:

____________________________________ ____

Borrower**
Approval from Loaning Unit (see instructions)

**I accept responsibility for reasonable care and security of all University property that is in my custody off campus in pursuit of my official duties. Liability assessment, if any, will be based on "Section 12.1 – Custodianship of Property" in *Business and Financial Policies and Procedures*.

______________ ______________________ ____________________________________

Date Returned Received by Phone #

This form may be photocopied.

Appendix 3

Mobile Learning Communities (MLC) Project, Spring 2010

'Exit' Questionnaire about the Usage of Google Phones in cs241 Class

The MLC team would appreciate if you fill out the questionnaire so that we can improve the phone usage/wireless services for the next generation of students. As you know, this MLC project was an experiment, funded by the National Science Foundation. Please, email the filled out questionnaire to klara@illinois.edu. All questionnaires will stay anonymous, i.e., all information about the sender will be strictly removed in the final processing of the questionnaire answers to the National Science Foundation according to the privacy guidelines of the University of Illinois.

1. **Have you used the leased Google phone during the period of March 30- May 14?** (yes/no)
2. **If you used the Google Android phone answer the following questions with respect to usage of the Mi-Clicker services in class (quiz, event notification, text/audio messaging):**
   a. Have you used the phone for quizzes? (yes/no)
   b. Have you used the phone for text messaging? (yes/no)
   c. Have you used the phone for notification of events? (yes/no)
   d. What was your experience with the Google Android interface/performance of the Mi-Clicker services? (Please, provide a constructive feedback so that we can improve on these services for the next generation of students.)
   e. Did you rely on any of the Mi-Clicker services (e.g., text-messaging or event notification)? Explain.
   f. What were the problems you saw (perceived) when using the Google Android phone inside the 1404 classroom that hindered you to use efficiently the Mi-Clicker services (e.g., authentication issues, power issues, login issues, user interface, ….)?
3. **If you used the Google Android phone outside of the classroom, please, respond to the following questions:**
   a. What other Google Android applications/services over WiFi networks did you use on the phone?
   b. Did you put your telephone SIM card into the Google Android phone and used it for telephone service? (yes/no)
   c. Did you program your own services on the Google Android phone? If yes, what kind of services/applications did you program?
   d. Did you attend the Google Android phone programming workshop this semester? (yes/no)
   e. f you attended the programming workshop, did you find it useful and should the workshop be offered more regularly during the next semesters? Explain.
4. **Future potentials of the Google Android phone platform:**
   a. Would you like to see more classes that use the Google Android platform in their classes? If yes, explain what classes you would like to see them in and for what purposes.
   b. What other in-classroom educational services on Google phones would you like see to be used by students, instructors and TAs?

c. What other outside-of-classroom services on Google phones would you like to see?

Any further suggestions what worked or did not work with respect to the Google Android Phones and you would wish that should be solve